

Universidad de San Carlos de Guatemala  
Facultad de Ingeniería  
Escuela de Ciencias y Sistemas  
Arquitectura de Computadores y Ensambladores 1  
Segundo Semestre de 2019  
Ing. Otto Escobar  
Tutor Académico Sección A: Ricardo Menchú  
Tutor Académico Sección B: Oscar Cuéllar



## Tarea Práctica 2 de Laboratorio

### Contenido

1 OBJETIVOS .....	2
1.2 OBJETIVOS ESPECIFICOS .....	2
2 EQUIPO Y MATERIAL UTILIZADO .....	3
3 DESCRIPCIÓN .....	3
3.1 Laberinto .....	3
3.2 Aplicación Android .....	4
3.3 Sistema de Colores .....	4
3.4 Modos del Sistema .....	4
3.4.1 Modo Automático .....	4
3.4.2 Modo Manual .....	4
3.4.3 Consideraciones .....	5
4 REFERENCIAS .....	5
4 ENTREGABLES .....	5
5 OBSERVACIONES Y RESTRICCIONES .....	6

# 1 OBJETIVOS

---

## 1.1 OBJETIVO GENERAL

- Que el estudiante adquiera, aplique e interactúe con microcontroladores.

## 1.2 OBJETIVOS ESPECIFICOS

- Comprender el funcionamiento de las entradas y salidas, tanto digitales como análogas de los microcontroladores.
- Aplicar el lenguaje C para estructuras de control en microcontroladores.
- Utilizar el módulo bluetooth.
- Realizar aplicaciones móviles que interactúen con el microcontrolador.
- Emplear la memoria EEPROM del microcontrolador.
- Manejar motores con el microcontrolador.
- Manejar sensores con el microcontrolador.

## 2 EQUIPO Y MATERIAL UTILIZADO

---

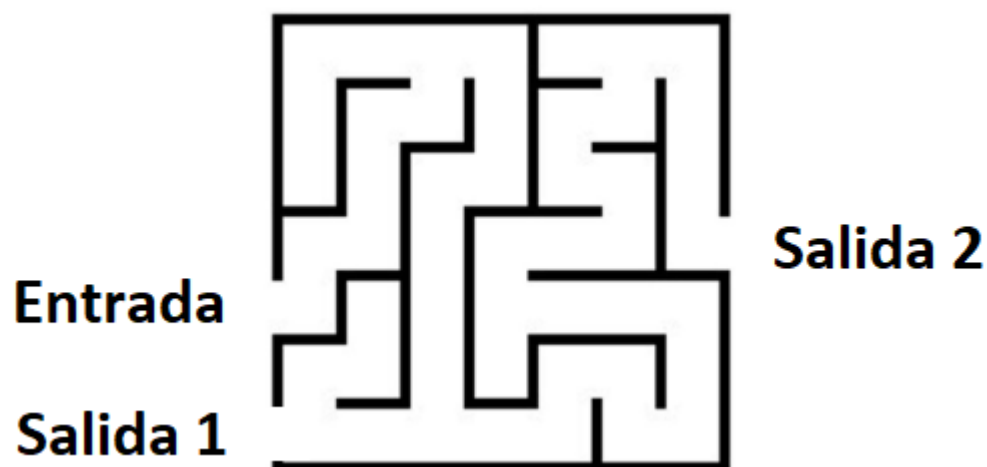
- Microcontrolador Arduino (MEGA - Recomendado)
- Sensor de color.
- Sensor ultrasónico.
- Módulo Bluetooth.
- Motores (Stepper, dc, servomotor).

## 3 DESCRIPCIÓN

---

La práctica consiste en realizar un vehículo controlado por una aplicación android y automático el cual viajará por un laberinto con dos posibles salidas. El camino para la primera salida estará indicado através de colores pegados en las paredes de cada muro los cuales serán detectados por el vehículo y con los cuales tomará la desición de que acción realizar. El camino para la segunda salida no tendrá ningún tipo de color.

### 3.1 Laberinto



La estructura del laberinto debe ser la que se muestra en la imagen.

### 3.2 Aplicación Android

Se debe contar con una aplicación en android con las siguientes funciones:

- \* Movimiento del vehículo. Atrás, Adelante, Izquierda, Derecha.
- \* Limpiar Ruta.
- \* Guardar Ruta.
- \* Recorrer Ruta Guardada.
- \* Recorrer Ruta Guardada Invertida.
- \* Modo del vehículo. Manual, Automático.

Estas funciones se explican mas a detalle luego.

### 3.3 Sistema de Colores

Los colores serán los encargados de cambiar la dirección a la que se mueve el vehículo.

Color	Función
Rojo	Detener movimiento.
Verde	Mover izquierda.
Azul	Mover derecha.

### 3.4 Modos del Sistema

#### 3.4.1 Modo Automático

En este modo, el vehículo comienza a desplazarse hacia adelante, el sensor ultrasónico mide la distancia a la que se encuentra el muro y deja el vehículo detenido en una posición favorable para que este pueda detectar con el sensor de color el color que hay en el muro (nunca pegado al muro). Luego de detectar el color, el vehículo toma un nuevo camino dependiendo del color antes leído.

#### 3.4.2 Modo Manual

En este modo el vehículo es capaz de moverse dependiendo de las instrucciones enviadas a través del módulo bluetooth, cada vez que se envía un movimiento, estos se van guardando en memoria SRAM del arduino, al momento de usar la opción Guardar Ruta la aplicación nos pedirá un nombre (máximo 4 caracteres) para la ruta, se guardará en memoria EEPROM del Arduino.

Al seleccionar la opción de “Recorrer Ruta Guardada” se leerán las 2 rutas guardadas en la memoria EEPROM con todo y su nombre. Al seleccionar una de ellas, el vehículo comenzará a desplazarse siguiendo la ruta que acaba de leer.

Al seleccionar la opción de “Recorrer Ruta Guardada Invertida” se leerán las 2 rutas guardadas en la memoria EEPROM con todo y su nombre. Al seleccionar una de ellas, el vehículo comenzará a

desplazarse siguiendo la ruta que acaba de leer en forma invertida, es decir, los últimos giros y desplazamientos que guardó, serán los primeros en realizarse.

La opción “Limpiar Rutas”, limpiará las rutas guardadas en la memoria EEPROM dejando las celdas vacías para poder ingresar una nueva ruta.

### 3.4.3 Consideraciones

\* Para comprobar que se esté utilizando la memoria EEPROM del Arduino correctamente, se reiniciará la placa y se conectará un nuevo dispositivo Android al vehículo (esto para evitar que se guarde en memoria del teléfono las rutas).

## 4 REFERENCIAS

1. Memorias en Arduino: <https://aprendiendoarduino.wordpress.com/2017/06/21/memoria-flash-sram-y-eeeprom-3/>
2. Memoria EEPROM: <https://www.luisllamas.es/guardar-variables-entre-reinicios-con-arduino-y-la-memoria-no-volatil-eeeprom/>  
<https://aprendiendoarduino.wordpress.com/2017/09/15/eeeprom-2/>

## 4 ENTREGABLES

---

- Código fuente de Arduino
- Manual de usuario y técnico

Nombre: [ARQ1]ManualUsuario\_Grupo#.pdf

Nombre: [ARQ1]ManualTecnico\_Grupo#.pdf

Entregar el archivo por medio de la plataforma **Classroom** antes de las 23:59 horas del jueves 22 de agosto de 2019. Se calificará el día siguiente a la entrega. Los horarios de calificación y la hoja de calificación serán publicados en los días próximos a la entrega.

Nombre: [ARQ1]P2\_G#S(A|B).zip | .rar

Ejemplo: [ARQ1]P2\_G5SB.rar

## 5 OBSERVACIONES Y RESTRICCIONES

---

- El proyecto debe realizarse en grupos no mayores a 4 personas.
- **La aplicación debe realizarse con Android Studio.**
- No es permitido prestarse el microcontrolador entre grupos (para evitar copias de código).
- Se tomará en cuenta en la nota la estética y buen diseño de los circuitos.
- El día de la calificación se harán preguntas sobre la elaboración de la práctica para comprobar que todos los integrantes hayan trabajado.
- Documentación copiada tendrá una penalización del 25% de la nota obtenida.
- Copias tendrán nota de 0 puntos y serán reportados al catedrático y a la escuela de ciencias y sistemas.
- Respetar el formato de nombre del entregable ya que este será utilizado para comparar copias entre ambas secciones.
- Se calificará del código entregado.